Volume 10 Number 1 1982 Nucleic Acids Research

Microcomputer programs for DNA sequence analysis

Bruce Conrad and David W.Mount

Department of Molecular and Medical Microbiology, College of Medicine, University of Arizona, Tucson, AZ 85724, USA

Received 15 September 1981

ABSTRACT

Computer programs are described which allow (a) analysis of DNA sequences to be performed on a laboratory microcomputer or (b) transfer of DNA sequences between a laboratory microcomputer and another computer system, such as a DNA library. The sequence analysis programs are interactive, do not require prior experience with computers and in many other respects resemble programs which have been written for larger computer systems (1-7). The user enters sequence data into a text file, accesses this file with the programs, and is then able to (a) search for restriction enzyme sites or other specified sequences, (b) translate in one or more reading frames in one or both directions in order to find open reading frames, or (c) determine codon useage in the sequence in one or more given reading frames. The results are given in table format and a restriction map is generated. The modem program permits collection of large amounts of data from a sequence library into a permanent file on the microcomputer disc system, or transfer of laboratory data in the reverse direction to a remote computer system.

INTRODUCTION

Computers are being widely used for storage and analysis of DNA sequences (1-8). Most of the programs previously described have been designed to run on large computers which have considerable memory space and are equipped to translate a variety of programs into a machine executable form. These programs enable the user to enter new sequences, to locate various types of symmetry within the sequence or homologies between the sequence and another (1-10), to generate restriction maps (10), to translate or to show various types of statistical data such as codon useage (1-7). Due to the huge volume of sequence data becoming available, computer data banks will play a important role in the future in giving a large number of users access to the data and supporting programs. The intent of this paper is to describe programs which allow (a) use of a microcomputer equipped with a telephone communications device (modem or accoustic coupler) to communicate with a remote computer, such as one with a DNA library bank, and exchange data with the remote system using permanent files, and (b) certain types of DNA sequence analysis on a

laboratory microcomputer.

PROGRAM DESCRIPTIONS

System requirements

The programs decribed below are designed to run on an 8080 or Z80 type microcomputer. The modem program works with both 300 and 1200 BAUD modems, which must be interfaced to a serial communications port on the microcomputer.

The programs all run under the CP/M operating system (Digital Research). DNA sequence files are entered into a file using any standard text editor. The modem program is written in 8080 assembly language and the remaining programs in the C programming language.

The modem communications program (MCOMM.ASM)

This program has four communication modes: local, talk, input and output. In the local mode, the user keyboard input goes only to the local computer and reflected by it back to the terminal screen; there is no data output to the modem. However, the modem is monitored for any input data, which is displayed on the screen. In the talk mode, keyboard characters go to the modem, which is also monitored for input. It is assumed that the remote computer will send back all the characters it receives. In this mode, the microcomputer is behaving as a terminal; all the modem communications are transparent to the local microcomputer. In input mode, the keyboard characters go out the modem, and all incoming modem data goes to the standard keyboard input entry point on the microcomputer, there to be echoed to the screen. This mode is used after using the local mode to open a file for data input from the keyboard, then switching to input mode to instruct the remote computer to transmit data. We describe below a program CRFILE.C which can be used to recover data in input mode, although any text editor can also be used. In output mode, the keyboard characters enter the microcomputer and control its activity, data from the microcomputer is transmitted out the modem to the remote computer, and modem input characters pass to the terminal screen. In this mode, a data file is first opened on the remote system, the O communication mode invoked, and the local system instructed to transmit data. Data files filled in the input and output modes are closed manually by the approprate end-of-file command after switching to the corresponding direct communication mode.

Switching MCOMM to its various modes is controlled by a special keyboard character, control-a. Whenever this character is typed, the program waits for another character to be typed on the keyboard. The second character sets the new communication mode, displays a help table or the current communication

Table 1 - use of mcomm and crfile to transfer a file

```
A>mcomm

MODEM COMMUNICATIONS PROGRAM
TYPE "CONTROL A" THEN "H" FOR COMMAND LIST
NOW IN "L" MODE FOR LOCAL I/O ONLY

A>^ATYPE COMMAND > T

$type remotefile^ATYPE COMMAND > L

A>crfile b:localfile
^ATYPE COMMAND > I

KILL LINE FEEDS? Y/N : N
LINE FEEDS NOW PASS

;THIS IS A TEST SEQUENCE FROM REMOTE SYSTEM
;IDENTIFICATION OF FILE FOLLOWS
TEST SEQUENCE
ATGATTGGCATTGGCGCGCGGGGGGAAAATCAGACATAGACCACATATTTTATTCGGAG
AAATTTTTTTTTTCCC2
;COMMENTS FOLLOW SEMICOLON
;1 AT END MEANS LINEAR, 2 MEANS CIRCULAR MOLECULE

$^ATYPE COMMAND > L

A>^Z
 saving file B:LOCALFILE
```

Note: The notation "^A" means that the user must type a control-a at the keyboard. This is not echoed to the screen, but is shown here for clarity.

mode. If control-a is typed twice, control-a itself is sent to the existing reciever device. Therefore, all available keyboard characters can be passed by the program. It is also easy and possible to change to another special character for convenience.

The create-file program (CRFILE.C)

This program allows the user to save incoming data from the modem to a discette file on the microcomputer. Its most important feature is that it automatically stops input from the remote computer when the available memory in the microcomputer has been filled, and then writes the data from memory to the discette file. It then automatically restarts the remote system again and repeats the procedure. This feature makes possible the transfer of DNA sequences much greater in length than available memory, without loss of any data. We have saved up to 135,000 characters at a time with this program, the

Table 2 - Sample output from the resenz program

```
B>resenz

                RESTRICTION ENZYME SEARCH PROGRAM
                ---------- ------ ------ -------


    This program will search a DNA sequence and report the locations in
the sequence which are recognized by the restriction enzymes listed in
the file 'res'.

Enter name of sequence file: test

Sequence: TEST
AluI        AGCT            No match
HpaII       CCGG            No match
MnlI        CCTC            No match
MnlI        GAGG            No match
FnuDII      CGCG             15,  17,  19
DpnI        GATC            No match
HhaI        GCGC             14,  16,  18
HaeIII      GGCC            No match
RsaI        GTAC            No match
TaqI        TCGA            No match
HinFIII     ATTCG            54
HinFIII     CGAAT           No match
EcoP1       AGACC            40
EcoP1       GGTCT           No match
...         ...
AsuII       TTCGAA          No match
EcoRI*      PPATQQ           62
XhoII       PGATCQ          No match
HaeII       PGCGCQ          No match
SauI        CCTNAGG         No match
BstEII      GGTNACC         No match
Tth111I     GACNNNGTC       No match
BglI        GCCNNNNNGGC     No match
HgiEII      ACCNNNNNNGGT    No match
EcoK        AACNNNNNNGTGC   No match
EcoK        GCACNNNNNNGTT   No match
EcoB        AGCANNNNNNNNTC  No match
EcoB        TGANNNNNNNNTGC  No match

Sequence: TEST
        10        20        30        40        50        60        70
ATGATTGGCATTGGCGCGCGCGGGGGGAAAATCAGACATAGACCACATATTTTATTCGGAGAAATTTTTT
             HhaI FnuDII               EcoP1         HinFIII EcoRI*
              FnuDII
               HhaI
                FnuDII
                 HhaI

        80        90       100       110       120       130       140
TTTTCCC

Do you wish to search another sequence? n
B>
```

Table 3 - Sample dialogue with the program dnatra

```
B>dnatra
          DNA TRANSLATION
          --- -----------


 Enter the name of the sequence file: test
 Sequence: TEST
 Do you want a full translation in both directions in all three reading
 frames? (yes or no) y
 Sequence: 'TEST'

        1 ATG ATT GGC ATT GGC GCG CGC GGG GGG AAA ATC AGA CAT AGA CCA CAT   48
          Met Ile Gly Ile Gly Ala Arg Gly Gly Lys Ile Arg His Arg Pro His
            .  Leu Ala Leu Ala Arg Ala Gly Gly Lys Ser Asp Ile Asp His Ile
           Asp Trp His Trp Arg Ala Arg Gly Glu Asn Gln Thr  .  Thr Thr Tyr

       49 ATT TTA TTC GGA GAA ATT TTT TTT TTC CC  77
          Ile Leu Phe Gly Glu Ile Phe Phe Phe Pro
           Phe Tyr Ser Glu Lys Phe Phe Phe Ser His
            Phe Ile Arg Arg Asn Phe Phe Phe Pro Met

 Sequence: 'TEST' (complimentary strand)

       77 GGG AAA AAA AAA ATT TCT CCG AAT AAA ATA TGT GGT CTA TGT CTG ATT   30
          Gly Lys Lys Lys Ile Ser Pro Asn Lys Ile Cys Gly Leu Cys Leu Ile
           Gly Lys Lys Lys Phe Leu Arg Ile Lys Tyr Val Val Tyr Val  .  Phe
            Glu Lys Lys Asn Phe Ser Glu  .  Asn Met Trp Ser Met Ser Asp Phe

       29 TTC CCC CCG CGC GCG CCA ATG CCA ATC AT   1
          Phe Pro Pro Arg Ala Pro Met Pro Ile Met
           Ser Pro Arg Ala Arg Gln Cys Gln Ser Trp
            Pro Pro Ala Arg Ala Asn Ala Asn His Gly

 Do you want another translation of TEST? y
Sequence: TEST
 Do you want a full translation in both directions in all three reading
 frames? (yes or no) n
 Forward or backward? (f or b) f
 Which reading frame do you want? (Starting at base 1, 2, 3 or all) 1
 Sequence: 'TEST'

        1 ATG ATT GGC ATT GGC GCG CGC GGG GGG AAA ATC AGA CAT AGA CCA CAT   48
          Met Ile Gly Ile Gly Ala Arg Gly Gly Lys Ile Arg His Arg Pro His

       49 ATT TTA TTC GGA GAA ATT TTT TTT TTC CC  77
          Ile Leu Phe Gly Glu Ile Phe Phe Phe Pro

 Do you want another translation of TEST? n
 Do you want to translate another sequence? n

 B>
```

Table 4 - Sample results from the seqstat program

```
B>seqstat
                SEQUENCE STATISTICS
                -------- ----------


Enter name of sequence file test
Sequence 'TEST':
At which base number do you want analysis to begin? 46
Analysis from base 46 to which base? 77
Number of codons analyzed: 10
TTT-Phe  2 (20.0) TTC-Phe  2 (20.0) TTA-Leu  1 (10.0) TTG-Leu  0 ( 0.0)
TCT-Ser  0 ( 0.0) TCC-Ser  0 ( 0.0) TCA-Ser  0 ( 0.0) TCG-Ser  0 ( 0.0)
TAT-Tyr  0 ( 0.0) TAC-Tyr  0 ( 0.0) TAA- .   0 ( 0.0) TAG- .   0 ( 0.0)
TGT-Cys  0 ( 0.0) TGC-Cys  0 ( 0.0) TGA- .   0 ( 0.0) TGG-Trp  0 ( 0.0)
CTT-Leu  0 ( 0.0) CTC-Leu  0 ( 0.0) CTA-Leu  0 ( 0.0) CTG-Leu  0 ( 0.0)
CCT-Pro  0 ( 0.0) CCC-Pro  0 ( 0.0) CCA-Pro  0 ( 0.0) CCG-Pro  0 ( 0.0)
CAT-His  1 (10.0) CAC-His  0 ( 0.0) CAA-Gln  0 ( 0.0) CAG-Gln  0 ( 0.0)
CGT-Arg  0 ( 0.0) CGC-Arg  0 ( 0.0) CGA-Arg  0 ( 0.0) CGG-Arg  0 ( 0.0)
ATT-Ile  2 (20.0) ATC-Ile  0 ( 0.0) ATA-Ile  0 ( 0.0) ATG-Met  0 ( 0.0)
ACT-Thr  0 ( 0.0) ACC-Thr  0 ( 0.0) ACA-Thr  0 ( 0.0) ACG-Thr  0 ( 0.0)
AAT-Asn  0 ( 0.0) AAC-Asn  0 ( 0.0) AAA-Lys  0 ( 0.0) AAG-Lys  0 ( 0.0)
AGT-Ser  0 ( 0.0) AGC-Ser  0 ( 0.0) AGA-Arg  0 ( 0.0) AGG-Arg  0 ( 0.0)
GTT-Val  0 ( 0.0) GTC-Val  0 ( 0.0) GTA-Val  0 ( 0.0) GTG-Val  0 ( 0.0)
GCT-Ala  0 ( 0.0) GCC-Ala  0 ( 0.0) GCA-Ala  0 ( 0.0) GCG-Ala  0 ( 0.0)
GAT-Asp  0 ( 0.0) GAC-Asp  0 ( 0.0) GAA-Glu  1 (10.0) GAG-Glu  0 ( 0.0)
GGT-Gly  0 ( 0.0) GGC-Gly  0 ( 0.0) GGA-Gly  1 (10.0) GGG-Gly  0 ( 0.0)

B>
```

limit being the capacity of the microcomputer discette. In Table 1, we demonstrate how MCOMM and CRFILE may be used together to save incoming data from a remote computer system.

The restriction site search program (RESENZ.C)

This program is the most sophisticated of the set in its structure and length. It first reads in a file which lists the known restriction enzymes and the sequences which they recognize and then creates a search tree that allows very fast matching of these sequences against the input DNA sequence (11-12). The enzymes are listed in a table which shows the number of the first base in the sequence where a match is found, and a map of the input DNA strand is printed with the name of the enzyme positioned under the first base matched. This search procedure is rapid, taking about 15 seconds to match the default table of about 100 restriction enzymes to a 1000 basepair sequence. If necessary, other strings to be matched may be substituted for the existing list of enzymes. The maximum DNA length which can be analyzed is set at 8192 base pairs, but this can be increased if there is sufficient memory available.

The program will analyze both linear and circular DNA molecules. An example of the output of the program is shown in Table 2.

The DNA translation program (DNATRA.C)

This program prompts the operator for an input file containing the sequence to be analyzed, and for information regarding the extent of translation wanted. The amino acids are printed in 3-letter abbreviated form, with Met bold-faced and a period used to indicate a chain termination codon. Forward and backward translation is possible in any of the individual 3 reading frames, or all of them combined. This latter option is useful for revealing open reading frames. We are presently inproving the program so that it will, if requested, report these open reading frames to the user. An example of the program output is shown in Table 3.

The codon useage program (SEQSTAT.C)

This program reports the use of each codon in an input DNA sequence and prompts the user for the first base to be analyzed and the reading frame. A typical session is shown in Table 4.

Availability of programs

A detailed description of the system and its requirements, and executable copies of the programs will be made available for non-profit use upon written request.

ACKNOWLEDGEMENTS

The authors thank Dr. David R. Hanson for the use of computer facilities in the Dept. of Computer Sciences, University of Arizona and Dr. A. Richard Kassander, Vice-president for Research, and Dr. Lee Jones, Provost, University of Arizona for financial support. We also thank Dr. D. Brutlag, the Molgen group and the SUMEX-AIM facility at Stanford University, Dr. M. Dayhoff of the Nucleic Acid Sequence reference system at Georgetown University, and Dr. Hugo Martinez, University of California at San Francisco for showing us their facilities, and Dr. John Little for his advice concerning our sequence analysis programs. Support was also provided by grants GM24496 from the National Institutes of Health and PCM-791-12059 from the National Science Foundation.

REFERENCES

1. Staden, R. (1977) Nucl. Acids Res. 4, 4037-4051.
2. Staden, R. (1978) Nucl. Acids Res. 5, 1013-1015.
3. Staden, R. (1980) Nucl. Acids Res. 8, 817-825.

4. Gingeras, T.R., Milazzo, J.P., Sciaky, D., Roberts, R.J. (1979) Nucl. Acids Res. 7, 529-545
5. Gingeras, T.K., and Roberts, R.J. (1980) Science 209, 1322-1328.
6. Queen, C.L. and Korn, L.J. (1980) Methods in Enzymology 65, 595-609.
7. Korn, L.J., Queen, C.L., and Wegman, M.N. (1977) Proc. Nat. Acad. Sci. 74, 4401-4405.
8. Sege, R., Soll, D., Ruddle, F.H., and Queen, C. (1981) Nucl. Acids Res. 9, 437-444.
9. Fuchs, C., Rosenvold, E.C., Honigman, A., and Szybalski, W. (1978) Gene 4, 1-23.
10. Schroeder, J.L., Blattner, F.R. (1978) Gene 467-474.
11. Rabin, M.O., and Scott, D. (1959) IBM J. Res. 3, 115-125.
12. Thompson, K. (1968) Comm ACM 11, 419-422.